Security Measures in the Design Procedure to Safeguard API Being Used Accurately During interactions with Applications

Veenababu Kannika Sherly

Research Analyst, SMRVD Security Solutions, India.

Abstract: API usage is rising and empowering businesses to build more dynamic applications. However, as they take advantage of these capabilities, organizations need to be aware of the potential security holes, close them timely and ensure that security is the number one priority. An unsecured API or application endpoint can serve as a gateway to the data centre by which attackers can effectively attack the backend and there is no silver bullet when it comes to its security. Hackers are sophisticated and are constantly looking into new ways to break down defenses and access valuable data.

Keywords: Priority; Attacks; Application endpoints.

1. Introduction

API authentication means determining that the client application has an identity that is allowed to use the API. API must be able to authenticate itself to the Apps which consume it. Likewise, when your API interacts with Servers, they must authenticate themselves to the API. Tokens should expire regularly to protect against replay attacks. Most enterprises will use an internal database or LDAP authentication store, though Oath may be a better option for highly public APIs.

Multi–factor Authentication (MFA) requires a user to use a one–time usage token they receive after authenticating with her credentials. The User may also have a digital key which is a token that the App can validate. When the App receives the token which it validates with the MFA Provider, it proceeds to consume your API. Tokens are usually issued with an expiration period and can be revoked.

Authorization is determining the scope of interaction allowed – that is, what actions and data the authenticated application has access to when using the API. This is typically best handled by application logic for e.g. using an access control framework, such as OAuth. However, it is best to augment this functionality using an API gateway. The following two ways are also used for defining level of authorization required by the user [1-11].

Static assignment of roles to Users based on the organizational groups to which they belong. Groups are role and App agnostic, they are purely business–level decisions. In RBAC an App uses roles to assign degrees of access to groups of Users which the role represents. Attribute–based Access Control (ABAC) aims to facilitate the dynamic determination of access control based on some sort of circumstantial information available at the time of the API call.

Developers often have a feature–driven mindset, where functionality has taken precedence over security. Unfortunately, in today's security landscape, vulnerabilities and threats lurk at every corner and have ever–growing consequences, so we have to turn this on its head. SSL/TLS encryption is mainstream and should be used for both public and internal APIs to protect against man in the middle attacks, replay attacks, and snooping. For external APIs the web server can handle this directly or a reverse proxy can be employed. A service mesh can be used for internal APIs libraries to add automatic encryption on top of service discovery and routing.

Developers must ensure that the API properly validates all input from the user to prevent XSS and SQL Injection. There are many ways to protect against these types of vulnerabilities including but not limited to "cleaning user input to prevent XSS", as well as preventing SQL Injection by "preparing statements with bind variables". Some attackers may try to overwhelm the API or trigger a buffer overflow vulnerability with large requests. These may be in the form of a large JSON body or even unusually large individual JSON parameters within the request. Abnormally large response may also be indicator of data theft. Create custom rules to track and block these suspicious requests. A web application firewall can automatically detect and blocks this type of input abuse.

Throttling is a means of controlling or limiting a client's access to your data. There are two key API throttles that will provide you with additional control & security for your APIs: IP–based throttling – Using IP–based throttling you can restrict the number of API calls made by a particular IP address. In addition, you could ensure that your API can only be accessed by a particular set of IP addresses.

Rate–limit throttling – Rate–limit throttling allows API requests to be made until a certain limit has been reached for a specific time period. By utilizing rate–limit throttling within your

API you can help to ensure that the database isn't overwhelmed by one particular client who may be misusing your interface. Building tests that don't represent real functional use – Performing tests without considering how the APIs will be consumed may be quicker in the short–term.

However, in doing so, you won't be testing across concerns, which could prevent you from uncovering and debugging potentially serious API issues. Custom API Rules – Build your own business logic rules for security, for e.g. a simple protection might be to identify your authentication token (in the HTTP header or in the JSON body) and require it to always be present to block and log any unauthenticated attempts. Another example would be to enforce the Content–Type header to be what is expected for your API (e.g. application/json) or block unused or non–public HTTP methods (e.g. PUT and DELETE) to further lock down the API.

2. Methods

Building API tests can be a bit of a solo act, but the minute a test is in your workflow, it requires the attention of different teams in your organization. If you set up test failure notifications to go to just you, you're adding time, effort and headaches to your workflow. Keys in URI – For some use cases, implementing API keys for authentication and authorization is good enough. However, sending the key as part of the Uniform Resource Identifier (URI) can lead to the key being compromised. As explained in IETF RFC 6819, It's safer to send API keys is in the message authorization header, which is not logged by network elements. As a rule of thumb, the use of the HTTP POST method with payload carrying sensitive information is recommended.

Geofencing – If your API is public, it might make sense to either block users from countries you don't do business with, or at least raise the risk score of entities that come from those countries. API Fuzzing Protection – You may have a combination of documented and undocumented features in your APIs. Attackers may attempt to map and exploit the undocumented features by iterating or fuzzing the endpoints. Install a web application firewall for application profiling and behavior tracking. L7 DOS Protection – You have protected the front–end of the API with rate–limiting, but the back–end services can still be exposed to Layer 7 denial of service [12-23]. Customize a web application firewall to ensure long–running queries and eventually blocked automatically.

Use Auditing and Logging – Auditing should never be skipped. Logging should be systematic and independent, and resistant to log injection attacks. Auditing should be used as a tool for detecting and proactively preventing attacks. Monitor add–on software carefully – One popular use of the interfaces is to allow third parties to write add–on apps for a platform [24-37]. A potential monster is such interfaces often give developers a high level of authorization rights. Hackers covet those privileges and will voraciously try to dig out such system vulnerabilities. Secure the exit gateways – Businesses need to set up another checkpoint on the way out of the network. Even If a hacker worms into the system and accesses confidential information, it has value only if the data can be moved out to their own systems.

In other words, if you miss a crook on the way in, you still can thwart him on the way out. Stack Trace – Many API developers become comfortable using 200 for all success requests, 404 for all failures, 500 for some internal server errors, and, in some extreme cases, 200 with a failure message in the body, on top of a detailed stack trace. A stack trace can potentially become an information leak which attackers can exploit by submitting crafted URL requests. It’s a good practice to return a “balanced” error object, with the right HTTP status code, with minimum required error message(s) and “no stack trace” during error conditions [38-50]. This will improve error handling and protect API implementation details from an attacker.

Consider Adding Timestamp in Request – Along with other request parameters, you may add a request timestamp as HTTP custom header in API request. The server will compare the current timestamp to the request timestamp, and only accepts the request if it is within a reasonable timeframe (1–2 minutes, perhaps). This will prevent very basic replay attacks from people who are trying to brute force your system without changing this timestamp.

3. Conclusion

APIs are a strategic necessity to give your business the agility, innovation and speed needed to succeed in today’s business environment. However, the financial incentive associated with this agility is often tempered with the fear of undue exposure of the valuable information that these APIs expose. With the rise of APIs also comes the potential for more security holes, meaning coders need to understand the risk to keep corporate and customer data safe. Goal of this article is to make developers understand design principles and security best practices, to protect their APIs from malicious activity. Potential threats can often be avoided by thinking critically about these practices and applying them to avoid breaches and help your business maximize its potential.

References

[1] Cárdenas, A.A.; Amin, S.; Lin, Z.S.; Huang, Y.L.; Huang, C.Y.; Sastry, S. Attacks against process control systems: Risk assessment, detection, and response. In Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security, Hong Kong, China, 22–24 March 2011.

[2] GOST-R. Risk Management. Risk Assessment Methods; ISO/IEC 31010-2011; International Organization for Standardization: Geneva, Switzerland, 2009.

[3] Purdy, G. ISO 31000:2009—Setting a new standard for risk management. Risk Anal. 2010, 30, 881–886.

[4] Vinod Varma Vegesna (2017). “Incorporating Wireless Sensor Networks and the Internet of Things: A Hierarchical and Security-Based Analysis,” International Journal of Current Engineering and Scientific Research, Volume-4, Issue-5, Pages 94-106, Available at SSRN: https://ssrn.com/abstract=4418110

[5] Vinod Varma Vegesna (2016). “Threat and Risk Assessment Techniques and Mitigation Approaches for Enhancing Security in Automotive Domain,” International Journal of Management, Technology And Engineering, Volume VI, Issue II, July-Dec 2016, Pages 314-331, Available at SSRN: https://ssrn.com/abstract=4418100

[6] Hamid Ali Abed Al-Asadi and et al., “ Critical Comparative Review of Nature-Inspired Optimization Algorithms (NIOAs), International Journal of Simulation: Systems, Science and Technology (IJSSST), 2020, 21(3), PP1-15

[7] Hamid Ali Abed Al-Asadi, (2022) “1st Edition: Privacy and Security Challenges in Cloud Computing A Holistic Approach" Intelligent Internet of Things for Smart Healthcare Systems, Scopus, Taylor @Francis, CRC Press. (Book Chapter: Enhanced Hybrid and Highly Secure Cryptosystem for Mitigating Security Issues in Cloud Environments), March 2022.

[8] Vinod Varma Vegesna (2015). “Incorporating Data Mining Approaches and Knowledge Discovery Process to Cloud Computing for Maximizing Security,” International Journal of Current Engineering and Scientific Research, Volume-2, Issue-6, Pages 118-133, Available at SSRN: https://ssrn.com/abstract=4418107

[9] Islam, S.; Fenz, S.; Weippl, E.; Mouratidis, H. A Risk Management Framework for Cloud Migration Decision Support. J. Risk Financial Manag. 2017, 10, 10.

[10] Islam, S.; Mouratidis, H.; Weippl, E.R. An empirical study on the implementation and evaluation of a goal-driven software development risk management model. Inf. Softw. Technol. 2014, 56, 117–133.

[11] Berg, H.-P. Risk management: Procedures, methods and experiences. Risk Manag. 2010, 1, 79–95.

[12] Peng, Y.; Lu, T.; Liu, J.; Gao, Y.; Guo, X.; Xie, F. Cyber-physical system risk assessment. In Proceedings of the Ninth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, Beijing, China, 6–18 October 2013.

[13] Vinod Varma Vegesna (2020). "Secure and Privacy-Based Data Sharing Approaches in Cloud Computing for Healthcare Applications," Mediterranean Journal of Basic and Applied Sciences, Volume 4, Issue 4, Pages 194-209, October-December 2020, doi: 10.46382/mjbas.2020.4409.

[14] Hamid Ali Abed Al-Asadi and et al., "Priority Incorporated Zone Based Distributed Clustering Algorithm For Heterogeneous Wireless Sensor Network", Advances in Science, Technology and Engineering Systems Journal Vol. 4, No. 5, PP. 306-313, 2019.

[15] Hamid Ali Abed Al-Asadi and et al., "A Network Analysis for Finding the Shortest Path in Hospital Information System with GIS and GPS, Journal of Network Computing and Applications (2020) 5: 10-22.

[16] Hamid Ali Abed Al-Asadi, et al., "Nature Inspired Algorithms multi-objective histogram equalization for Grey image enhancement", Advances in Computer, Signals and Systems (2020) 4: 36-46 Clausius Scientific Press, Canada DOI: 10.23977/acss.2020.040106.

[17] Vinod Varma Vegesna (2019). "Investigations on Different Security Techniques for Data Protection in Cloud Computing using Cryptography Schemes", Indo-Iranian Journal of Scientific Research, Volume 3, Issue 1, Pages 69-84, January-March 2019, Available at SSRN: https://ssrn.com/abstract=4418119

[18] Vinod Varma Vegesna (2018). "Analysis of Artificial Intelligence Techniques for Network Intrusion Detection and Intrusion Prevention for Enhanced User Privacy", Asian Journal of Applied Science and Technology, Volume 2, Issue 4, Pages 315-330, Oct-Dec 2018, Available at SSRN: https://ssrn.com/abstract=4418114

[19] Cardenas, A.; Amin, S.; Sinopoli, B.; Giani, A.; Perrig, A.; Sastry, S. Challenges for securing cyber physical systems. In Proceedings of the Workshop on Future Directions in Cyber-Physical Systems Security, Newark, NJ, USA, 23–24 July 2009.

[20] Gai, K.; Qiu, M.; Zhao, H.; Tao, L.; Zong, Z. Dynamic energy-aware cloudlet-based mobile cloud computing model for green computing. J. Netw. Comput. Appl. 2016, 59, 46–54.

[21] Gai, K.; Qiu, M. Blend arithmetic operations on tensor-based fully homomorphic encryption over real numbers. IEEE Trans. Ind. Inform. 2017.

[22] Patel, S.C.; Graham, J.H.; Ralston, P.A. Quantitatively assessing the vulnerability of critical information systems: A new method for evaluating security enhancements. Int. J. Inf. Manag. 2008, 28, 483–491.

[23] Vinod Varma Vegesna (2021). “A Highly Efficient and Secure Procedure for Protecting Privacy in Cloud Data Storage Environments,” International Journal of Management, Technology and Engineering, Volume XI, Issue VII, July 2021, Pages 277-287.

[24] Hamzah F. Zmezm, Hareth Zmezm, Mustafa S.Khalefa, Hamid Ali Abed Al-Asadi, "A Novel Scan2Pass Architecture for Enhancing Security towards E-Commerce," Future Technologies Conference 2017, 29-30 November 2017 | Vancouver, BC, Canada, 2017.

[25] Hamid Ali Abed Al-Asadi, Majida Ali Al-Asadi, Nada Ali Noori , "Optimization Noise Figure of Fiber Raman Amplifier based on Bat Algorithm in Optical Communication network," International Journal of Engineering & Technology, Scopus, Vol 7, No 2, pp. 874-879, 2018.

[26] Hareth Zmezm, Mustafa S.Khalefa, Hamid Ali Abed Al-Asadi, Hamzah F. Zmezm, Dr. Hussain Falih Mahdi, Hassan Muhsen Abdulkareem Al-Haidari. "Suggested Mechanisms for Understanding the Ideas in Authentication System," International Journal of Advancements in Computing Technology9(3):10-24, 2018.

[27] Vinod Varma Vegesna (2021). “The Utilization of Information Systems for Supply Chain Management for Multicomponent Productivity Based on Cloud Computing,” International Journal of Management, Technology and Engineering, Volume XI, Issue IX, September 2021, Pages 98-113.

[28] Vinod Varma Vegesna (2021). “The Applicability of Various Cyber Security Services for the Prevention of Attacks on Smart Homes,” International Journal of Current Engineering and Scientific Research, Volume-8, Issue-12, Pages 14-21.

[29] Hahn, A.; Ashok, A.; Sridhar, S.; Govindarasu, M. Cyber-physical security testbeds: Architecture, application, and evaluation for smart grid. IEEE Trans. Smart Grid 2013, 4, 847–855.

[30] Ten, C.-W.; Manimaran, G.; Liu, C.-C. Cybersecurity for critical infrastructures: Attack and defense modeling. IEEE Trans. Syst. Man Cybern. Part A Syst. Hum. 2010, 40, 853–865.

[31] Sridhar, S.; Hahn, A.; Govindarasu, M. Cyber–physical system security for the electric power grid. Proc. IEEE 2012, 100, 210–224.

[32] Ray, P.D.; Harnoor, R.; Hentea, M. Smart power grid security: A unified risk management approach. In Proceedings of the 2010 IEEE International Carnahan Conference on Security Technology (ICCST), San Jose, CA, USA, 5–8 October 2010.

[33] Vinod Varma Vegesna (2022). "Utilising VAPT Technologies (Vulnerability Assessment & Penetration Testing) as a Method for Actively Preventing Cyberattacks," International Journal of Management, Technology and Engineering, Volume XII, Issue VII, July 2022, Pages 81-94.

[34] Majda Ali Abed and Hamid Ali Abed Al-Asadi, "Simplifying Handwritten Characters Recognition Using a Particle Swarm Optimization Approach", European Academic Research, Vol 1,pp. 535- 552, Issue(5), 5. 2013.

[35] Majda Ali Abed and Hamid Ali Abed Al-Asadi, "High Accuracy Arabic Handwritten Characters Recognition using (EBPANN) Architecture," International Journal of Advanced Computer Science and Applications (IJACSA), Vol. 6 Issue 2, 2015.

[36] Hamid Ali Abed Al-Asadi and Majda Ali Abed, "Object Recognition Using Artificial Fish Swarm Algorithm on Fourier Descriptors," American Journal of Engineering, Technology and Society; Volume 2, Issue 5: pp. 105-110, 2015.

[37] Vinod Varma Vegesna (2022). "Accelerate the development of a business without losing privacy with the help of API Security Best Practises - Enabling businesses to create more dynamic applications," International Journal of Management, Technology and Engineering, Volume XII, Issue IX, September 2022, Pages 91-99.

[38] Vinod Varma Vegesna (2021). "Analysis of Data Confidentiality Methods in Cloud Computing for Attaining Enhanced Security in Cloud Storage," Middle East Journal of Applied Science & Technology, Vol. 4, Iss. 2, Pages 163-178, April-June 2021, Available at SSRN: https://ssrn.com/abstract=4418127

[39] Yadav, D.; Mahajan, A.R. Smart Grid Cyber Security and Risk Assessment: An Overview. Int. J. Sci. Eng. Technol. Res. 2015, 4, 3078–3085.

[40] Yoneda, S.; Tanimoto, S.; Konosu, T.; Sato, H.; Kanai, A. Risk Assessment in Cyber-Physical System in Office Environment. In Proceedings of the 2015 18th International Conference on Network-Based Information Systems (NBiS), Taipei, Taiwan, 2015.

[41] Georgieva, K.; Farooq, A.; Dumke, R.R. Analysis of the Risk Assessment Methods–A Survey. In International Workshop on Software Measurement; Springer: Berlin, Germany, 2009.

[42] Vinod Varma Vegesna (2022). "Using Distributed Ledger Based Blockchain Technological Advances to Address IoT Safety and Confidentiality Issues," International Journal of Current Engineering and Scientific Research, Volume-9, Issue-3, Pages 89-98.

[43] Hamid Ali Abed Al-Asadi, Majida Ali Abed, AL-Asadi, Zainab sabah, Baha Al-Deen, Ahmad Naser Ismail, "Fuzzy Logic approach to Recognition of Isolated Arabic Characters", International Journal of Computer Theory and Engineering, Vol. 2, No. 1, 1793-8201, 2010.

[44] H. A. Al-Asadi, M.H. Al-Mansoori, S. Hitam, M. I. Saripan, and M. A. Mahdi, "Particle swarm optimization on threshold exponential gain of stimulated Brillouin scattering in single mode fibers," Optics Express, vol. 19, no. 3, pp. 1842-1853, 2011.

[45] Majida Al-Asadi, Yousif A. Al-Asadi, Hamid Ali Abed Al-Asadi, "Architectural Analysis of Multi-Agents Educational Model in Web-Learning Environments," Journal of Emerging Trends in Computing and Information Sciences, Vol. 3, No. 6, 2012.

[46] Vinod Varma Vegesna (2022). "Methodologies for Enhancing Data Integrity and Security in Distributed Cloud Computing with Techniques to Implement Security Solutions," Asian Journal of Applied Science and Technology, Volume 6, Issue 2, Pages 167-180, April-June 2022, doi: 10.38177/ajast.2022.6217.

[47] Cherdantseva, Y.; Burnap, P.; Blyth, A.; Eden, P.; Jones, K.; Soulsby, H.; Stoddart, K. A review of cyber security risk assessment methods for SCADA systems. Comput. Secur. 2016.

[48] Gai, K.; Qiu, M.; Ming, Z.; Zhao, H.; Qiu, L. Spoofing-jamming attack strategy using optimal power distributions in wireless smart grid networks. IEEE Trans. Smart Grid 2017.

[49] Rice, E.B.; AlMajali, A. Mitigating the risk of cyber attack on smart grid systems. Procedia Comput. Sci. 2014, 28, 575–582.

[50] ISO. Risk Management—Principles and Guidelines; ISO 31000:2009; International Organization for Standardization: Geneva, Switzerland, 2009.